

WHAT IS CLAIMED IS:

- 1 1. A method for computing distances between a received point and four points in a two-
2 dimensional grid with a constellation representing a number of bits greater than three, wherein
3 each of the four points belong to a unique coset in the constellation, the method comprising:
4 determining a first point on a grid nearest to the received point;
5 computing a second point closest to the received point inside a specified area;
6 computing a third, fourth, and fifth point, wherein each point is a member of a different
7 coset and each point is the closest point in its coset to the received point; and
8 computing a distance from the received point to each of the second, third, fourth, and
9 fifth points.
- 1 2. The method of claim 1 further comprising after the first computing, recomputing the
2 second point if the second point is invalid.
- 1 3. The method of claim 2, wherein the second point is invalid if it is outside of the
2 constellation.
- 1 4. The method of claim 1, wherein the first point can be determined by evaluating:
2 $\text{round}((R_x + iR_y - 1 - i)/2*2 + 1 + I,$
3 wherein R_x and R_y are two-dimensional components of the received point, i is the imaginary
4 number, and $\text{round}(\cdot)$ is an operator that returns an integer number closest to a value provided to
5 it.
- 1 5. The method of claim 1, wherein the number of bits is an even value, wherein the received
2 point can be expressed in two-dimensional components R_x and R_y , and wherein the first

3 computing comprises:
4 determining if Rx and Ry lie inside a square specified by the number of bits; and
5 computing two-dimensional components of the second point based on the second
6 determining.

1 6. The method of claim 5, wherein the second determining comprises:
2 setting $C_x = 1$ if Rx lies inside a boundary of the square, else $C_x = -1$;
3 setting $C_y = 1$ if Ry lies inside a boundary of the square, else $C_y = -1$;
4 and wherein the fourth computing comprises
5 setting $A_x = \text{sign}(\text{RG}_x) * \text{MAX}_{XY}$ if $C_x = -1$, else $A_x = \text{RG}_x$; and
6 setting $A_y = \text{sign}(\text{RG}_y) * \text{MAX}_{XY}$ if $C_y = -1$, else $A_y = \text{RG}_y$,
7 wherein A_x and A_y are two-dimensional components of the second point, RG_x and RG_y are
8 two-dimensional components of the first point, MAX_{XY} is value describing the size of the square
9 and can be computed by $2^{\text{number of bits}/2} - 1$.

1 7. The method of claim 5, wherein the second computing comprises:
2 computing an intermediate value, d, wherein $d = \text{the received point} - \text{the second point}$;
3 setting the third point = the second point + $C_x * \text{sign}(dx) * 2$;
4 setting the fourth point = the second point + $i * C_y * \text{sign}(dy) * 2$; and
5 setting the fifth point = the second point + $2(C_x * \text{sign}(dx) + i * C_y * \text{sign}(dy))$,
6 wherein C_x and C_y are values specifying if the two-dimensional components of the received
7 point lie inside a boundary of the square and dx and dy are two-dimensional components of d.

1 8. The method of claim 5, wherein the third computing comprises computing a Euclidean
2 distance from the received point to each of the second, third, fourth, and fifth points.

1 9. The method of claim 8, wherein each of the second, third, fourth, and fifth points belong
2 to a unique coset.

1 10. The method of claim 1, wherein the number of bits is an odd value greater than three,
2 wherein the received point can be expressed in two-dimensional components Rx and Ry, and
3 wherein the first computing comprises computing two-dimensional components of the second
4 point, wherein the second point lies within a first square.

1 11. The method of claim 10, wherein the first square encompasses a cross-shaped
2 constellation, wherein a value MAX_{XY} describes the size of the first square and can be computed
3 by expression $(2^{\lceil \text{number of bits}/2 \rceil} + 2^{\lfloor \text{number of bits}/2 \rfloor})/2 - 1$, wherein a second square that is a largest
4 square encompassing 2^N constellation points with N being an even integer that can be enclosed
5 in the first square, wherein MAX_X and MAX_Y describe the size of the second square, and
6 wherein the first computing comprises:
7 setting $A_x = \text{sign}(RG_x) * MAX_{XY}$ if $\text{abs}(RG_x) > MAX_{XY}$ else $A_x = RG_x$; and
8 setting $A_y = \text{sign}(RG_y) * MAX_{XY}$ if $\text{abs}(RG_y) > MAX_{XY}$ else $A_y = RG_y$,
9 wherein A_x and A_y are two-dimensional components of the second point, and RG_x and RG_y are
10 two-dimensional components of the first point.

1 12. The method of claim 10 further comprising after the first computing, recomputing the
2 second point if the second point is invalid.

1 13. The method of claim 12, wherein the second point is invalid if $(\text{abs}(R_x) > MAX_X$ and
2 $\text{abs}(R_y) > MAX_Y)$ is true.

1 14. The method of claim 12, wherein the recomputing comprises
2 setting the second point = $Ax + i * \text{sign}(Ay) * MAX_Y$
3 and a sixth point = $\text{sign}(Ax) * MAX_X + i * Ay$ if $\text{abs}(Rx) > \text{abs}(Ry)$
4 else setting the second point = $\text{sign}(Ax) * MAX_X + i * Ay$
5 and the sixth point = $Ax + i * \text{sign}(Ay) * MAX_Y$,
6 wherein Ax and Ay are two-dimensional components of the invalid second point and α is
7 another point in the two-dimensional grid.

1 15. The method of claim 14, wherein the second computing comprises:
2 determining if the second point lies on the boundaries of the first square and α lies on the
3 boundaries of a second square, wherein the second square is the largest square encompassing 2^N
4 constellation points with N being an even integer that can be enclosed in the first square, wherein
5 MAX_X and MAX_Y describe the size of the second square; and
6 computing the third, fourth, fifth points.

1 16. The method of claim 15, wherein the second determining comprises:
2 setting $Cx = -1$ if $((\text{abs}(Rx) \geq MAX_X) \text{ and } (\text{abs}(Ay) > MAX_Y)) \text{ or } (\text{abs}(Rx) \geq$
3 $MAX_X))$ is true, else $Cx = 1$;
4 setting $Cy = -1$ if $((\text{abs}(Ry) \geq MAX_Y) \text{ and } (\text{abs}(Ax) > MAX_X)) \text{ or } (\text{abs}(Ry) \geq$
5 $MAX_Y))$ is true, else $Cy = 1$;
6 setting $C'x = 1$ if $R'x$ lies within the boundaries of the second square else $C'x = -1$; and
7 setting $C'y = 1$ if $R'y$ lies within the boundaries of the second square else $C'y = -1$,
8 wherein $R'x$ and $R'y$ are the two-dimensional components of α .

1 17. The method of claim 15, wherein the second computing comprises:
2 computing an intermediate value, d , wherein $d = \text{the received point} - \text{the second point}$;
3 setting the third point = the second point + $C_x * \text{sign}(dx) * 2$;
4 setting the fourth point = the second point + $i * C_y * \text{sign}(dy) * 2$; and
5 setting the fifth point = the second point + $2(C_x * \text{sign}(dx) + i * C_y * \text{sign}(dy))$,
6 wherein C_x and C_y are values specifying if the two-dimensional components of the received
7 point lie inside a boundary of the constellation and dx and dy are two-dimensional components
8 of d .

1 18. The method of claim 17, wherein the second computing further comprises:
2 setting a seventh point = the sixth point + $C'_x * \text{sign}(d'_x) * 2$; and
3 setting an eighth point = the sixth point + $i * C'_y * \text{sign}(d'_y) * 2$,
4 wherein d'_x and d'_y are the two-dimensional values of a second intermediate value, d' , wherein
5 $d' = \text{the received point} - \text{the sixth point}$, and C'_x and C'_y are values specifying if the two
6 dimensional components of the sixth point lie inside a boundary of the second square.

1 19. The method of claim 18, wherein the second computing further comprises checking to
2 ensure that the fifth point is a valid point.

1 20. The method of claim 19, wherein the validity of the fifth point can be checked and
2 corrected using expression:

3 if ($\text{abs}(\text{REAL}(D)) > \text{MAX}_X$) and ($\text{abs}(\text{IMAG}(D)) > \text{MAX}_Y$)
4 { if ($\text{abs}(dx) > \text{abs}(dy)$) then
5 $D = D - 4 * \text{sign}(\text{IMAG}(D)) * i$,
6 else

7 $D = D - 4 * \text{sign}(\text{REAL}(D)) \}$,

8 wherein D is the fifth point, and dx and dy are the two-dimensional components of the fifth
9 point.

1 21. The method of claim 18, wherein the third computing comprises computing a Euclidean
2 distance from the received point to each of the second, third, fourth, fifth, sixth, seventh, and
3 eighth points.

1 22. The method of claim 21, wherein each of the second, third, fourth, fifth, sixth, seventh,
2 and eighth points belongs to one of four unique cosets, and wherein in a coset with more than
3 one point, points with larger Euclidean distances to the received point are discarded.

1 23. The method of claim 1, wherein the method can be used to decode a received point in a
2 communications system.

1 24. The method of claim 23, wherein the communications system is an asymmetric digital
2 subscriber line (ADSL) compliant system.

- 1 25. A method for computing distances between a received point and four points in a two-
2 dimensional grid with a constellation representing a number of bits equal to three, wherein each
3 of the four points belong to a unique coset in the constellation, the method comprising:
4 computing a point from each coset to the received point; and
5 computing a distance from the received point to the point in each coset.
- 1 26. The method of claim 25, wherein the first computing comprises:
2 setting a first point = $1 + i$ if $R_x > -1$, else the first point = $-3 + i$;
3 setting a second point = $-1 - i$ if $R_x < 1$, else the second point = $3 - i$;
4 setting a third point = $1 - i$ if $R_y < 1$, else the third point = $1 + 3i$; and
5 setting a fourth point = $-1 + i$ if $R_y > -1$, else the fourth point = $-1 - 3i$;
6 wherein R_x and R_y are the two component values of the received point.
- 1 27. The method of claim 25, wherein the second computing comprises computing a
2 Euclidean distance between the received point and the point in each coset.